BEEBLET

ISSN 0112-3351

CONTENTS

Editorial 2
BBC+ Unveiled 3
Regional News 4
HELP 5
Short Listings 8
Voltage/current characteristics 11
Sort Report II 14
Access Radio 22
Some SAS concepts 24
BBC Lectern 27
Reviews: Acid Drops, Bird Strike 29
New magazine lister 31
A Refutation 33
Review - Using BBC BASIC 37
Who got their money back? 38
Hall of Fame 39
Local Groups 40

VOLUME 3 NUMBER 5 JUNE 1985

BBC/ACORN
COMPUTER USER GROUP
NZ (inc)

EDITORIAL

Once again its that grey, uninspiring time of the morning - I'm almost out of paper, double sided sellotape, and coffee. The twink is unusable. In short, another isue of Beeblet is about to rise from the ashes!

I am still looking for software COMPETITION entries. See the last few issues for details.

According to the April Editorial in Beebug, Acorn have stated that "the life of the BBC micro system is already assured for at least another four years through the contract with the BBC, and the deal with Olivetti can only improve future support and development."

ERRORS

There were at least two errors in the May listings :p.11 col. 2, line 460 (the "Gravity" listing) is obviously incorrect but it is too hard to work out what it should have been at this time in the morning. Similarly there is something wrong with the listing on page 35 - I'm working on it too.

The vertical broken line character (sharing the \ key on the BBC B keyboard) is defined by Wordwise as a pad character. Thus listings spooled to wordwise, then printed, have spaces substituted for this symbol - unfortunate in the case of function key definitions especially. One example of this is David Baker's speech utility program published in March (cf p7, lines 60 and 70 - it should precede the M). I normally add them to the listings by hand.

BEEBLET is edited by Bruce Wills who may sometimes be tracked down on 861-662 (h) or 729-929 (wk).

DISCLAIMER

The views of the reviewers, the Editor and other contributors are not necessarily shared by the BBC/ACORN COMPUTER USER GRO
OF NZ INC.

The BBC B PLUS Unveiled

Neil Gordon

At long last, in early May, Acorn Computers launched their upgraded version of the BBC Model B - the B Plus. It has a new circuit board, with the following features:

* An additional 32K RAM. This seems to work like Watford's 32K RAM board, in that 20K of it can be used for the screen display, with the additional 12K for printer or disk buffer space, or whatever.

* The disk interface and DFS is included as standard. This uses the Western Digital 1770 disk controller chip (as used by Solidisk and others), instead of the expensive and rare 8271. It can also be used for double density, but I don't know if Acorn have implemented this as an option. (They haven't -Ed.) The DFS is on a 16K ROM and now includes formatting and verification commands, etc., but doesn't appear to implement the Advanced Disk Filing System as used in the Electron Plus 3.

* An extra 2 ROM slots have been put in, to make a total of 6. These can now take 32K ROMs, and the BASIC and OS have been combined on one 32K ROM.

That's the good news. The bad news is that it will sell in Britain for 499 pounds (including VAT). Come on, Acorn. You can buy a BBC B with 32K RAM card from Watford for about 405 pounds, add the Solidisk double density DFS for 40 pounds, and a Watford sideways ROM board for about 30 pounds, to get a similarly equipped computer for 475 pounds. While it's nice to have it all in a neat package with the B Plus, this is an indication of the apparently ripoff prices that Acorn are asking. With rapidly falling chip prices, it must surely be cheaper to build a B Plus now than it was to build a B three years ago, implying that the B Plus should have been priced at no more than 400 pounds, and preferably less, with remaining stocks of the B dropping to 300 pounds or less. Even at 400 pounds, the B Plus would be around twice the price of the Commodore 64 and Amstrad, perceived (erroneously) by the public to be of similar power.

The BBC B, B Plus, and Acorn Electron are excellent computers. However, with Acorn apparently determined to price themselves out of the market, and in spite of the Acorn contract with the BBC, I fear for their future. The prices must drop.

NEWS FROM THE REGIONS

WELLINGTON

Don Churchill is the new President of the Wellington branch (see back cover). Workshops are now planned, on an experimental basis, to run on the second Thursday of each month. The June workshop is about printers, a follow up to the May meeting.

The winner of the 2YB software contest was Julian Wright with a touch typing tutor. Runner-up was a Mr Speirs, Palmerston North?

AUCKLAND

Steven Williams reports that the May meeting was something of a non-event, however firm plans were made for June and July. Dave Smith was scheduled to speak in June about the use of a Beeb for sharemarket analysis. July is slotted for a discussion of expansion options (eg. Aries vs Solidisk vs co-processor etc). Other ideas are sought for future meetings.

Steve would like to know "who the local Solidisk expert in Manawatu is".

WAIKATO

..are now circulating their own newsletter, "Waibug".

TAURANGA

Jane C. admits to being "quite chuffed" after a major Kiwi fruit exporter aquired a Beeb to run a stock control program that she and Chris have been developing. They have almost finished one for the coolstores to use and hope to demonstrate it at the fruit grower's feild day in Te Puke in November.

What have the rest of you been doing?

COMPUTER CONCEPTS COMPILER

Steven W. reports "The news is that its out, and takes up two ROMs (one 16K, one 8K). There are two compile options: Semi-compiled (ie you need the ROMs resident to run the code) or full machine code. The pnly snag is that although the semi compiled format supports all BASIC statements except EVAL, the full compiler supports only integer arithmetic. They have also released a ROM which will let you use your ACORN speech upgrade on a phoneme basis ie construct your own words. Very useful.

HELP

A forum conducted by Tim Ryan

Q1 I was asked at the last local user group meeting why a program was crashing with the 'No PROC' message at the ENDPROC line of a procedure although there was a DEF PROC above.

A1 All BASICs use a stack to hold REPEAT/UNTIL, FOR/NEXT, PROC/ENDPROC information. Acorn have done theirs a bit different from most, in that there is a separate stack for each of the different types of loop (I'm including PROC/ENDPROC here).
Due to the different areas for each type, it is up to the user to make sure that the closure of the loops is done in the correct order.

```
Eg FOR      \
   REPEAT    \
   UNTIL      > in a program is OK
   NEXT      /

   FOR      \
   REPEAT    \
   NEXT       > in a program may give odd results
   UNTIL     /
```

The program below is designed to show one possible error that is caused by getting the loop closures out of order.
When execution of the PROC drops through to the ENDPROC, the REPEAT/UNTIL loop has not been closed. BASIC then executes a different UNTIL that is nothing to do with the original REPEAT within the PROCedure. The result of that is to jump back to the last REPEAT seen, and continue executing from there (as in a normal REPEAT/UNTIL loop).
Only trouble is, BASIC thinks that it has already finished the PROCedure (remember, we executed the ENDPROC statement before). So when it hits the ENDPROC for the second time, it gets horribly confused.
If there was a PROCedure that called PROCtest, instead of an error message and a bombed program, BASIC would begin executing

in the outer PROCedure which would produce totally unpredictable results.
TRY THE PROGRAM ! I bet you can't guess what it does.

```
   10 CLS
   20 PRINT "Go through PROCedure"
   30 PROCtest
   40 PRINT "Just before UNTIL"
   50 UNTIL FALSE
   60 PRINT "End of routine"
   70 END
   80
   90
  100 DEF PROCtest
  110
  120 REM The REPEAT is started inside the
  140 REM procedure, but the first UNTIL met is executed
  160 REM AFTER the ENDPROC. This UNTIL may be nothing
  170 REM to do with the original REPEAT
  180
  190 REPEAT
  200 PRINT "Should get this far"
  210 GOTO 230
  220 UNTIL FALSE
  230 ENDPROC
  240 PRINT "Should not get here"
```

So remember: When using loop structures, always finish them correctly to prevent all sorts of unrelated (and therefore impossible to debug) problems.
If the program logic requires you to jump out of a loop (and due to the lack of a DO/WHILE structure it does happen), there is a tidy way to do it. What I do is simply set a flag, and make the loop test for that flag.

Eg:
```
1230 IF (error condition) THEN error=TRUE:GOTO 1260
1240 ......
1250 ......
1260 UNTIL (normal end condition) OR error
```

Eg:
```
20 FOR T=1 TO 100
30 IF (error condition) THEN T=100:GOTO 50
40 ........
50 NEXT T
```

Q2 This is one I worked out myself, but I guess the answer may interest others. What I wanted to do was turn HEX numbers into strings, so that I could format the printout using RIGHT$ etc.

A2 It took me a little fiddling, but I finally came up with the answer:
Use the following

```
string_variable$=STR$~decimal_number
string_variable$=STR$~(decimal_number)
```

```
Eg: hexval$=STR$~23
    hexval$=STR$~(32*56/value1%)
```

where ~ is the tilde character (above the ^ on the keyboard).

HINT: It is useful to put HEX numbers into strings for formatting printed output.
Use : `PRINT RIGHT$("000"+hexval$,4);`
to print out hex numbers as four digits.
What all the string handling does is simply to pad the numeric out to four places with zeroes if the original number was less than 1000.

HINT: To all the Electron owners. You must remember that the Electron does not have Mode 7, therefore any program designed to use Mode 7 will not work correctly. If the Mode 7 command is only used for instructions, at the start of the program, and all the rest is in Mode 2 (for example) then all is well. However, if the whole program runs in Mode 7 then you are in trouble.

```
Eg: 10 MODE 7
    20 PROCinstructions
    30 MODE 2
    40 PROCsetup
    50 PROCgame
    60 END
```

Will probably work fine

```
Eg: 10 MODE 7
    20 PROCgame
    30 END
```

Will not work.

MORE SHORT LISTINGS

Neil B. Auckland

Editor's note: I have still a number of Neil's delightful short listings to publish. These can easily be further extended. More such programs would be welcome, as would themes for others to tackle.

PLUGRAPH

This is useful for graphing a number of related curves on a common vertical scale against (say) a time scale, to give an immediate appreciation for comparision and trends. The vertical scale is automatically set to accomodate the largest value input. A printer dump would be a useful addition to the program.

CONVERT

An easily extended menu of measurement conversions (eg. fahrenheit to and from centigrade) is presented. To return to the menu, press RETURN instead of inputing a value - in the case of temperature conversions, ANY letter key followed by RETURN.

SINLAND

Based on a listing for another machine (PROCsky added). This draws landscapes with random sine curves. The drawing process is halted (after the current curve is completed) by pressing the space bar. Useful additions would be colour, and a screen-dump routine for those with the appropriate equipment.

```
 10 REM::PLUGRAF::N.E.B. 14/3/85
 20 REM DRAWS A MULTIPLE LINE STATIST
    ICS GRAPH
 30 MODE0
 40 VDU23,8202;0;0;0;
 50 MAX=0
 60 INPUT''"NO. OF CURVES ",C
 70 INPUT'"NO. OF POINTS ",P
 80 DIM V(C,P)
 90 FOR CURVE=1TOC
100   CLS
110   PRINT''"CURVE #";CURVE
120   FOR point=1TO P
130     PRINT'"VALUE ";point
```

```
 10 REM::CONVERT:: NEIL B. 10/84
 20 MODE7
 30 @%=10
 40 DIM A$(20),B$(20),C(20)
 50 PROClist
 60 PROCselect
 70 PROCheader
 80 REPEAT
 90   IF A$(N)="deg.C" PROCtemp ELSE
      PROCcalc
100   UNTIL FALSE
110
120 DEF PROClist
130 PRINT
```

```
140     INPUT V(CURVE,point)
150     IF MAX<V(CURVE,point) MAX=V(C
        URVE,point)
160     NEXT
170   VDU7
180   NEXT
190 PROCaxes
200 PROCvert
210 PROCscale
220 PROChor
230 PROCplot
240 VDU4,10,10
250 END
260
270 DEFPROCplot
280 HFAC=1000/(I%*10)
290 FOR CURVE=1 TO C
300   X=60
310   MOVEX,V(CURVE,1)*HFAC
320   FOR point=1 TO P
330     Y=V(CURVE,point)*HFAC
340     DRAWX,Y
350     IF point=P MOVE 1265,Y+15:PRI
        NT;CURVE
360     X=X+INC%
370     NEXT
380   NEXT
390 ENDPROC
400
410 DEF PROCaxes
420 CLS:VDU5
430 MOVE 60,1000
440 DRAW 60,0
450 DRAW1260,0
460 ENDPROC
470
480 DEFPROCvert
490 INC%=1200/(P-1)
500 FOR across=1TO P
510   MOVE60+(across*INC%),0
520   PLOT21,60+(across*INC%),1000
530   NEXT
540 ENDPROC
550
560 DEF PROCscale
570 I%=.9+MAX/10
580 F%=1
```

```
140 X=0
150 REPEAT
160   X=X+1
170   READ A$(X),B$(X),C(X)
180   PRINT;X;TAB(3)A$(X);TAB(19)B$(X
    )
190   UNTIL A$(X)=""
200 VDU11:PRINT"  "
210 ENDPROC
220
230 DEF PROCselect
240 REPEAT
250   INPUT"Select number required "N
260   UNTIL N>0 AND N<X
270 ENDPROC
280
290 DEFPROCheader
300 CLS
310 PRINTA$(N);TAB(20)B$(N)'
320 VDU28,0,24,38,2
330 ENDPROC
340
350 DEF PROCcalc
360 INPUT"enter value "Q
370 IF Q=0 RUN
380 VDU11
390 @%=10:PRINT;Q;:@%=&20409:PRINTTAB
    (17)"=";TAB(20);Q/C(N)
400 @%=&20409:PRINT;Q*C(N);:@%=10:PRI
    NTTAB(17)"=";TAB(20);Q'
410 ENDPROC
420
430 DEF PROCtemp
440 @%=&20109
450 INPUT"enter temp "Q$
460 IF ASC(Q$)>58 RUN: REM ENTERING A
    NY LETTER WILL RETURN TO THE MENU
470 Q=VAL(Q$)
480 VDU11
490 PRINT;Q;TAB(17)"=";TAB(20);(Q*1.8
    )+32
500 PRINT;(Q-32)/1.8;TAB(17)"=";TAB(2
    0);Q'
510 ENDPROC
520
530 REM ADD YOUR OWN FAVOURITE CONVER
    SIONS BELOW
```

```
590 REPEAT
600    IF I%>9 I%=.9+(I%/10):F%=F%*10
610   UNTIL I%<10
620 IF I%>2 AND I%<5 I%=5
630 IF I%>5 I%=10
640 I%=I%*F%
650 ENDPROC
660
670 DEF PROChor
680 FOR UP=1TO10
690   MOVE0,15+(UP*100)
700    IF UP MOD 2=0 PRINT;UP*I%
710   MOVE60,UP*100
720   PLOT21,1260,UP*100
730   NEXT
740 ENDPROC
```

```
540 DATA millimetres,inches,25.4
550 DATA metres,feet,.3048
560 DATA kilometres,miles,1.6093
570 DATA sq.metres,sq.feet,.0929
580 DATA kilograms,pounds,.4535
590 DATA grams,ounces,28.35
600 DATA litres,gallons,4.456
610 DATA pascals,ins w.g.,249.1
620 DATA kilopascals,PSI,6.895
630 DATA litres/sec,CFM,2.12
640 DATA watts,HP,746
650 DATA joules,BTU,1055
660 DATA deg.C,deg.F,0
670 DATA "","",0
```

```
 10 REM::SINLAND::BYTE SEPT 84
 20 REM MODIFIED AND ADAPTED FOR BBC
    BY NEIL B. 1/85
 25 REM USERS WITH COLOUR SCREENS COU
    LD ADD COLOUR
 30 *KEY0 RUN|M
 40 MODE0
 50 PROCsky
 60 REPEAT
 70   phase=RND(1000)
 80   len=RND(300)+100
 90   axis=RND(470)+50
100   hfac=axis:IF axis>300 hfac=300-
      (axis-300)
110   height=RND(hfac-9)+10
120   step=RND(7):IF step<>2 step=RND
      (4)+2
130   A=RND(3):IF A=3 col=7 ELSE col=
      5
140   FORX=0TO1274 STEPstep
150     Y=SIN((X+phase)/len)*height+a
        xis
160     MOVEX,axis
170     PLOTcol,X,Y
180     NEXT
190   UNTIL INKEY(0)=32:REM PRESS SPA
    CE BAR TO ARREST DEVELOPMENT
200 END
210
220 DEFPROCsky
230 LOCALstep
240 step=44
250 T=1024
260 REPEAT
270   MOVE0,T:PLOT21,1278,T
280    T=T-step
290    step=step-2
300    UNTILstep<4
310 ENDPROC
```

For any of you wondering how I produce the solid line at the top and bottom of the page, I put my Citoh printer into 7 bit mode (using the appropriate DIP switch), then put a string of 's in a Define Heading (DF) and Define Footing (DF) command, in which the printer's ' bit graphics table is also invoked. I could probably call the graphics character directly in 8 bit mode...

VOLTAGE - CURRENT CHARACTERISTICS

Don F.

There are four channels at the Analogue Port which will measure voltages in the range 0 to 1.8 volts. The converter changes these to a number in the range 0 to 65520.

This program uses two of these channels to measure the voltage and current of a device such as a lamp, resistor or diode.

In order to convert the current to a voltage input, a suitable resistor is put in series with the lamp. In a more elaborate version of the program an inverting amplifier would be connected across the resistor wnich enables the voltage to be fed to the computer independently. In the simplest form of the program described here, the voltage across the resistor is subtracted from the voltage across both to give the voltage across the lamp.

This resistor must be chosen to give a voltage of less than 0.9V at full current.

It is advisable to protect the computer from excess voltages by connecting a zener diode across each input. I have an analogue input box which contains the amplifier, potential dividers and zener diodes. Most components will need a higher voltage than the computer can cope with, so a potential divider must be used to bring it down to 2.8V. This is shown in the diagram.

```
 10 REM DECAY SERIES
 20 REM D.Fraser 5/83
 30 MODE 1
 40 REPEAT
 50   VDU26:CLS
 60   A=600:B=0:T=0
 70   A1=0:B1=0
 80   C=0:D=0
 90   C1=0:D1=0
100   VDU19,0,4,0,0,0
110   PROCinput
120   VDU 23,1,0;0;0;0;
130   VDU29,0;250;
140   PROCframe
150   *FX 15,0
160   VDU19,0,4,0,0,0
```

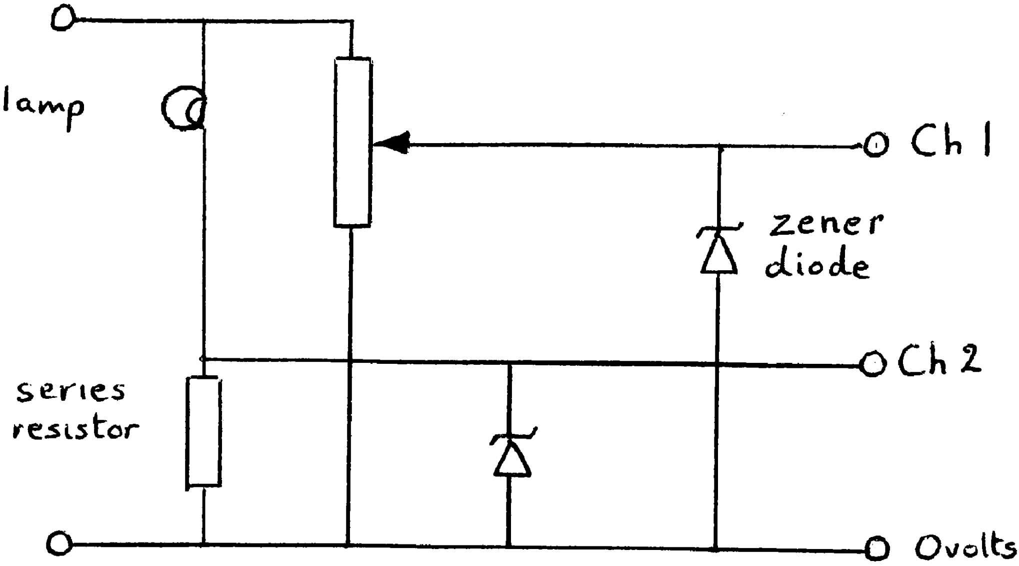
```
170   REPEAT
180     PRINT TAB(5,29)"Press space b
        ar to stop"'"
                           "

190     REPEAT
200       @%=&003
210       PRINT TAB(34,5)T
220       @%=10
230       PROCDraw(A1,A,200,1)
240       PROCDraw(B1,B,400,2)
250       PROCDraw(C1,C,600,3)
260       PROCDraw(D1,D,800,1)
270       PROCcalculate
280       R$=INKEY$(0)
290       UNTIL R$=" "
300     PROCwait
310     UNTILQ$="S" OR Q$="s"
320   UNTILFNstop
330 END
340
350 DEFPROCinput
360 PRINT TAB(5,10)"What is the half
    life of the"
370 INPUT TAB(5,12)"first element 5-1
    000 " E
380 IF E<1 OR E>1000 THEN 370
390 INPUT TAB(5,16)"Second element  5
     - 1000 " F
400 IF F<1 OR F>1000 THEN 390
410 INPUT TAB(5,20)"Third element 5 -
     1000 " G
420 IF G<1 OR G>1000 THEN 410
430 PRINT TAB(5,24)"The last element
    is stable"
440 R=INKEY(40)
450 ENDPROC
460
470 DEFPROCDraw(Y1,Y2,X,col%)
480 IF Y2>Y1 THEN GCOL0,col% ELSE GCO
    L 0,0
490 MOVE X,Y2
500 MOVE X+100,Y2
510 PLOT 85,X,Y1
```

```
520 PLOT 85,X+100,Y1
530 ENDPROC
540
550 DEF FNstop
560 VDU28,0,31,39,29:CLS
570 PRINT TAB(10,30)"Do you wish to r
    erun?"
580 REPLY$=GET$
590 =(REPLY$="N")OR(REPLY$="n")
600
610 DEFPROCframe
620 GCOL0,2:CLG
630 MOVE 110,-10
640 DRAW 1000,-10:DRAW 1000,610
650 DRAW 110,610:DRAW 110,-10
660 PRINT TAB(10,2)"Half life"
670 @%=&20006
680 PRINT TAB(4,4)E,F,G,"  Stable"
690 PRINT TAB(10,27)"Number of atoms"
700 PRINT TAB(33,3)"Time"
710 ENDPROC
720
730 DEFPROCcalculate
740 A1=A
750 A=A-A/E
760 B1=B
770 B=B+A/E-B/F
780 C1=C
790 C=C+B/F-C/G
800 D1=D
810 D=D+C/G
820 FOR J=1 TO 50:NEXT
830 A%=A:B%=B:C%=C:D%=D
840 PRINT TAB(18,25),D%:PRINT TAB(12,
    25)C%
850 PRINT TAB(6,25)B%:PRINT TAB(0,25)
    A%
860 T=T+1
870 ENDPROC
880
890 DEFPROCwait
900 PRINTTAB(5,29)"Press 'C' to conti
    nue, "'"              'S' to stop"
910 Q$=GET$
920 ENDPROC
```

The first part of the program asks for the value of this resistor and the ratio of voltage drop at the potential divider. If the voltage is now turned up to maximum, the computer can measure current and voltage in order to draw the axes of the graph. When the voltage has been turned down again the main program can begin. The graph is plotted "warts and all" giving irregularities which seem to be faults in the system, but as they are repeatable, an explanation can be found: The resistance of a lamp increases with temperature, so that as the voltage is increased the resistance lags behind its true value for that voltage. However on maintaining a constant voltage, the lamp heats up and the current drops after a short time. For this reason, the graph plotted for a decreasing voltage is lower than the one plotted for an increasing voltage.

Other components which can be compared are resistors, diodes, and transistors.
A possible extra feature for the program would be the ability to store the graphs and then superimpose them.



SORT REPORT - PART TWO

By Steven R.

The bubble sort improves...

This is the second in the series about sorting. In this article, I will explain both the INSERT sort, and the BINARY INSERT sort, and relegate the BUBBLE sort to last position.

It has been pointed out to me that most people want a "black box" procedure which sorts. They don't want to know how it works, but only what information it requires to work. I agree, but this type of series can be helpful when you are "boldly going where no-one has gone before" and would like an idea of how to improve what you have. If you follow from one of these sorts to the next, you can see how one was improved to become the next, as with the insert sort, which is an improved bubble sort.

Also, most importantly, it gives our Editor something to put in the newsletter...

All programs from now on are converted from working PASCAL programs. There may be a more efficient way of coding them, but they should work as shown. I haven't tested them exhaustively though, so they may have an insect or two (bug) in them from the conversion. Let the trier beware...

INSERT SORT

This sort is basically an improved bubble sort, where each item is inserted once into it's correct place, but many passes must be made.

Like the bubble sort, it only needs one pass to decide whether the array is sorted or not, and therefore the sorted array is it's best case (The best case is the one in which the method works best. The worst case is the one where it does the worst).

A typical procedure (formatted by me) looks like this:

```
100 DEF PROC_INSERT_SORT(maxitems)
110 first = element(1)
120 position = 1
130 FOR entry = 2 TO maxitems
140    IF element(entry) < first THEN
```

```
              first = element(entry):position = entry
150 NEXT
160 element(position) = element(1)
170 element(1) = first
180 FOR entry = 3 TO maxitems
190    current = element(entry)
200    position = entry
210    IF current < element(position - 1) THEN
              REPEAT:
                  element(position) = element(position - 1):
                  position = position - 1:
              UNTIL current >= element(position - 1)
220    element(position) = current
230 NEXT
240 ENDPROC
```

Lines 100 - 150 find the lowest element in the array, and put it's position in the variable 'position'.

Lines 160 and 170 swap the element just found with the first element in the array. This makes the first (top) element the lowest in the array.

The FOR - NEXT loop in lines 180 to 230 goes through the remaining elements in the array and inserts them into their correct place.
This is done as follows:

Line 190 saves the time taken in looking up the array element several times (and in typing it), by putting the current element in the variable 'current'.

Line 200 puts 'entry' in the variable 'position', because it may be changed at line 210. While using 'entry' in the loop works (I've tried it), it is bad practice to change the loop control variable inside the loop.

Line 210 simulates a WHILE - ENDWHILE loop by doing an IF <condition> THEN REPEAT UNTIL NOT <condition>. That's why the line is so long. The lack of WHILE is where BBC BASIC falls down a little. Still, one manages with what one has...

If 'current' is less than the element at 'position', then the element at 'position' is replaced with the element one higher, at 'position' minus one, shuffling the higher element down one slot. We now move our "frame of reference" up one place and try again.

This is repeated until the correct place for 'current' has been found, at which time 'current' is put into the array, at line 220.

ADVANTAGES

1. This sort is faster than the bubble sort.
2. It sorts arrays that are already in order almost as quickly as the bubble sort.
3. No extra room is needed besides the array.

DISADVANTAGES

1. (See ADVANTAGES) This sort is not much faster than the bubble sort, because although it does less comparisons, it still does a lot of passes through the array, and shuffles items about more than the bubble sort.

BINARY INSERT SORT

This is an improvement on the insert sort, where the insertion point is found by doing a binary search instead of a serial (sequential) search. Before I go any further, I'll explain the:

Binary Search

A binary search does something like what you do when you look up a phone number in the phone directory. First, the list to be searched MUST be sorted. Now, imagine that the list is in ascending order, with the smaller items at the top. Okay ?

The binary search routine first looks in the middle of the list. If it hasn't hit the jackpot, then it sees if the item in the list is bigger or smaller than what it is looking for. If the item in the list is bigger, then the one it wants must be higher, so it picks a point halfway between the top of the list and the place where it is now, and tries again. If the item in the list is smaller, then it has to try again lower down. It picks a point halfway down the portion of the list below where it is, and tries again.

To keep track of where it should be looking, it keeps two markers: top and bottom. When it decides to look higher, it sets the bottom marker to it's current position, or the top marker if looking lower. Then it finds the spot half-way between the two markers, and that becomes the current position. By setting markers like this, the area that is being searched gets halved every try, because if an area of the list has been looked at already, then the top or bottom marker will be set above or below that place, and it won't ever be looked at again in the search.

It keeps searching like this until either it finds what it's looking for, or the markers meet, which means that the item wasn't there. This is one of the fastest searches, and can be orders of magnitude faster than a serial search if the list is long.

If you are lucky, I'll give you a procedure to do a binary search next time. Meanwhile, to make sure that you understand how the routine works, why not have a look at the program listing below, and see if you can come up with the original, unmodified binary search yourself ?

Now, coming back to the sort...

The main way in which the program structure differs from the insert sort is that while the insert sort shuffles numbers up the list until the insertion point is found, the binary insert sort finds the insertion point first, and then shuffles the numbers up the list.
This really means that there are now two separate operations instead of one larger one.

The binary insert program looks like this:

```
 10 first = element(1)
 20 position = 1
 30 FOR entry = 2 to maxitems
 40    IF element(entry) < first THEN
             first = element(entry):
             position = entry
 50 NEXT
 60 element(position) = element(1)
 70 element(1) = first
 80
 90 FOR entry = 3 TO maxitems
100    current = element(entry)
110    found = FALSE
120    lo = 1
130    hi = entry - 1
140    REPEAT
150      position = INT(lo + hi) / 2  ←  = INT((lo+hi+1)/2)
160      IF current < element(position) THEN
              hi = position - 1
170      IF current > element(position) THEN
```

```
                lo = position + 1
 180     IF current = element(position) THEN
                lo = position + 1:
                found = TRUE
 190     IF hi < lo THEN
                found = TRUE
 200    UNTIL found
 210
 220    position = entry
 230    IF position < lo THEN                 >
            REPEAT:
               element(position) = element(position - 1):
               position = position - 1:
            UNTIL position >= lo          <=
 240    element(position) = current
 250 NEXT
```

Lines 10 - 70 are the same as in the insert sort.

Lines 90 - 200 find the insertion point by using a slightly modified binary search.

Lines 120 - 130 set the markers to the top and bottom of the sorted part of the list.

Lines 140 - 200 find the insertion point.

Line 150 calculates the place to try.

Lines 160 - 180 adjust the markers, depending on whether the insert point is higher or lower than the current spot.

Line 190 checks to see if the markers have crossed over. If they have, then its time to stop. This should only happen if the item is to be inserted at the top of the list, which should never happen because the lowest item was placed at the top in lines 10 - 70. I'm just covering all the possibilities here - 'should' does not necessarily mean 'will' !

Lines 220 - 240 shuffle the numbers down the list so that the item can be inserted at the correct place.

ADVANTAGES

1. This sort is consistantly faster than the insert sort, unless the data is already sorted, where it is slower than both of the other sorts. This is because the binary search is so much faster than a serial search. When the data is already sorted,

however, the binary search takes longer to find the insertion point than the serial search, slowing it down.

2. As with the other two sorts, no extra room is needed besides the array.

DISADVANTAGES

The only disadvantage compared to the other two sorts is that the program is longer than both. You usually know if the array is sorted before you start, and you wouldn't bother sorting it unless you had to.

Next time: an even better, faster sort.

CLASSIFIED ADS
****

DEALS AND DISCOUNTS

Send stamped addressed envelope for details of the deals negotiated for UG members. Please mark your request for the attention of the Equipment Officer, C/O the UG Box 9592 Wgtn.

FOR SALE:

CASSETTE TAPES

C10s, boxed, leaderless - available from the UG in multiples of 20 : $30-00 plus $2-00 p&p.

MICRO COVERS

Designed to fit the BBC micro. Attractive opaque reinforced plastic, $5-50 posted. To Iain, Box 26078, Epsom, Auckland 3.

FLOPPY DISCS

3M Brand 5 1/4" DSDD discs - available from the UG ex-stock in (cardboard) boxes of 10: $55-00, which includes p&p

UNDOCUMENTED *FX CALLS

David B. supplies a few more undocumented calls:
*FX202,x is used to read or write the keyboard status, so -
*FX202,255 turns off the CAPS LOCK
*FX202,144 turns off CAPS LOCK and turns on SHIFT LOCK
*FX202,239 turns the CAPS LOCK back on.

Some PLOT commands are also unrecorded - David notes "I believe PLOT 77 fills something but I can't remember what."

The BBC Prestel Adapter

Simply and easily connected to a BBC Model B Microcomputer, the Prestel Adaper converts the computer into a powerful terminal capable of two way communication via telephone lines. Provides access to armchair shopping, Business information etc.

Features.

1. The Presetl Adapter enables users to communicate, through the Post Office Telephone Network, with various information services and databases in NZ and Overseas.

2. The Aditel System for example provides subscribers with a wide range of constantly updated information.

3. Prestel allows the BBC Model B to "talk" to almost any computer with a 1200/75 Bits per second modem.

4. Available databases include: Share and stock prices, weather, news , wool and livestock prices, farming information etc etc. More databases are constantly being added.

5. The Prestel Adapter allows the user to edit text until satisfactory and then send the completed messages by telephone. The system is fast, inexpensive and secure.

6. Information received from Databases can be saved to disc or cassette or dumped to a printer.

Who Needs The Prestel Adapter?

Anyone with a need for access to a wide variety of information, particularly farmers, travel agents etc. Prestel is a simple, inexpensive way of joining an rapidly expanding network of databases.

Why settle for less than the ultimate Videotex system

If you're presently using a Videotex terminal, or considering a new purchase, ask yourself two questions.

	Yes	No
(1) Can I use my dedicated terminal for anything other than receiving Videotex?	☐	☐
(2) Can I store Videotex information for fast recovery without the cost of going back on line?	☐	☐

With the BBC micro and its inexpensive Prestel* Adapter you can say yes to both these questions and enjoy the additional benefits of the powerful BBC system.

BBC PRESTEL

BARSON computers

For further informaton contact
Barson Computers NZ Ltd
1 Ngaire Avenue, Epsom,
P.O. Box 26-287, Auckland.
Phone (09) 504-049

---

*Prestel is a registered trademark of British Telecom
*British Broadcasting Corporation is abbreviated to BBC throughout the text

THE BBC MICROCOMPUTER
PRESTEL ADAPTER

AVAILABLE AT WHITCOULLS
AND ALL MAJOR BBC
RETAILERS

RETAIL PRICE $599-00

ADITEL SUBSCRIPTIONS
$180 PER ANNUM
+ 17c per minute

The PRESTEL adapter is a fully type approved 1200/75 baud direct autodialling modem and is supplied with PRESTEL software on ROM

For more information contact
BARSON COMPUTERS
1 NGAIRE AVE
EPSOM
AUCKLAND
Ph. (9) 504-049

ACCESS RADIO BROADCASTS

comment and hints! from

JOHN ANDREW-

INTRODUCTION

Access radio started broadcasting BBC MICRO programs two months back. Some users (including myself) had no success in receiving acceptable signals. However, in co-operation with Gary Clark of Access Radio, Tel (04)721777, an attempt has been made to get to the bottom of the problem. Hopefully the hints given below will lead to an increased interchange of ideas and progams via this medium.

TRANSMITTER CHARACTERISTICS

Access radio broadcasts on 783 kHz at 20 kW from Titahi Bay, just North of Wellington. Transmission of computer programs is scheduled at 9 A.M. on Saturdays. This timing is based on programming commitments and because it is a quiet period of the day insofar as electrical interference is concerned. (The alternate time of the early morning hours doesn't appear to have any great following!). BBC MICRO broadcasts are scheduled each month (see May BEEBLET page 16).

COMPUTER REQUIREMENTS

The graphs opposite show the sound level intensity plotted against frequency. To satisfactorily decode the signals the computer needs significant peaks at 1200 Hz and 2400 Hz and the harmonics at 3600, 4800 and 7200 Hz.

BROADCAST OF 20 APRIL

The upper curve opposite shows the signals received on 20 April at Silverstream some 15 km from the Transmitter. On that occasion a BBC MICRO and a colour TV were ON. The signals are buried deep in noise which explains why they couldn't be decoded by the MICRO. The compartively high level of low frequency (50 Hz and 100 Hz) could also cause signal suppression. Adjusting for maximum bass cut can suppress this interference, (data recorders do this anyway !)

BROADCAST OF 18 MAY

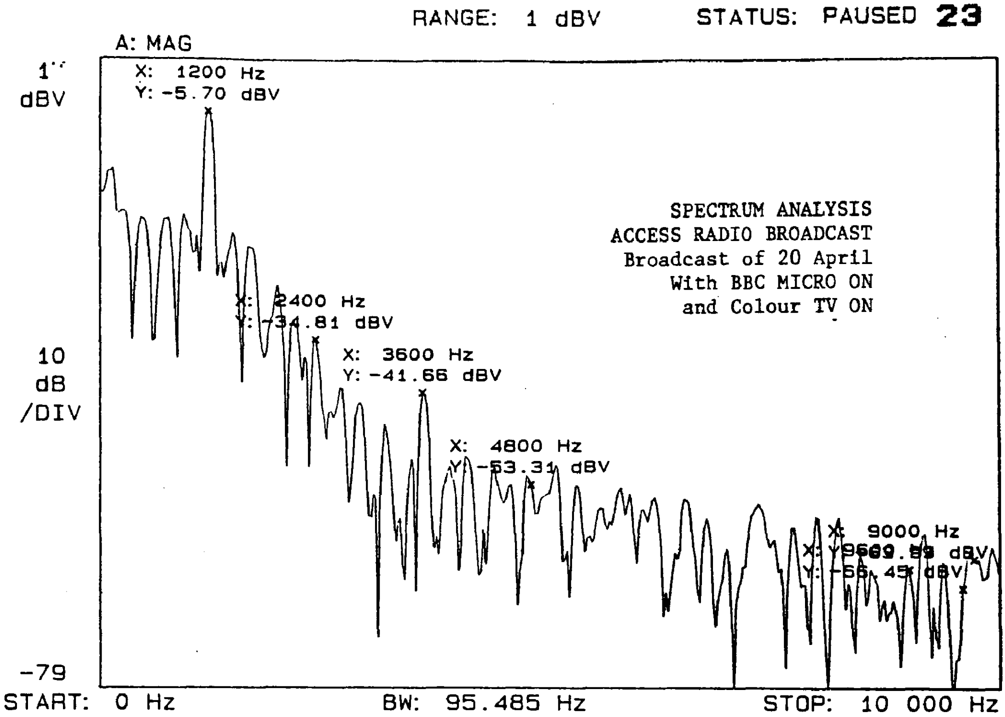
With the BBC MICRO and colour TVs turned OFF the signals were received on 18 May as shown in the lower graph. All blocks were decodable as the necessary peaks are well clear of noise.

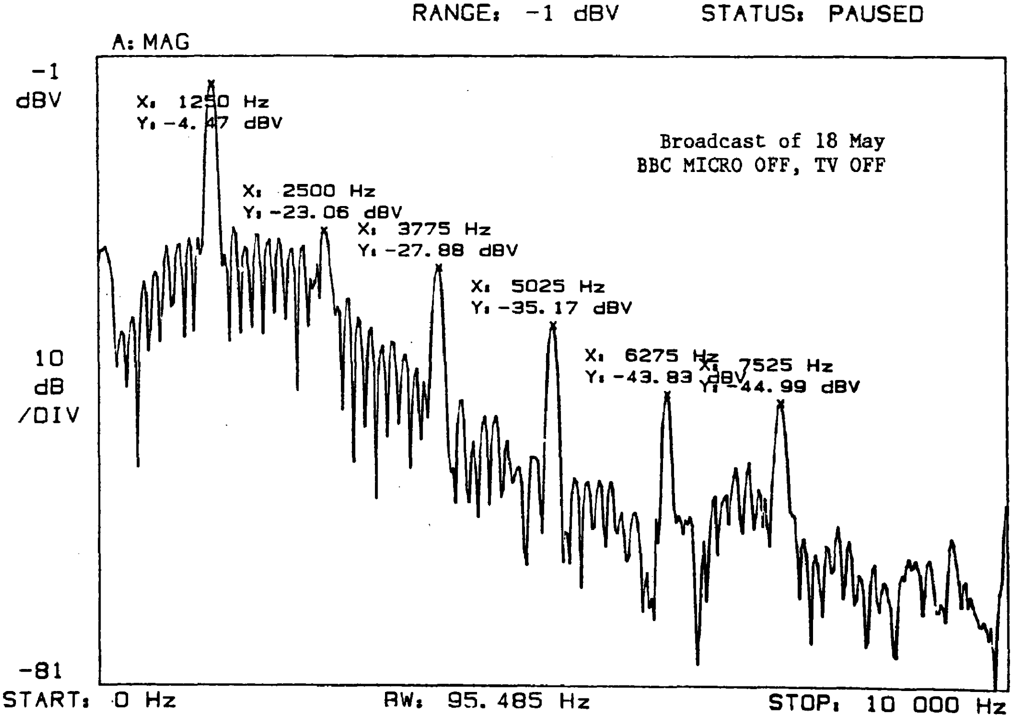
TAPE RECORDER POINTS

Data cassette recorders are optimised for storage of data. If using other recorders any limiters should be disabled. (Dolby systems could be used to advantage but if you dont understand the system turn it off.) The recording level can be safely set a few dB into the red as the signal level will be constant and not likely to overdrive the tape head.

HINTS FOR SUCCESSFUL RECEPTION

1. Turn off the BBC MICRO
2. Turn off any colour TV sets
3. Turn off any simmerstat or thermostat appliances (irons, hotplates)
4. Turn off limiters on the tape recorder
5. Set level a few dB into the red
6. Set bass cut to maximum


RANGE: 1 dBV
STATUS: PAUSED
A: MAG
1
dBV
10
dB
/DIV
-79
X: 1200 Hz
Y: -5.70 dBV
X: 2400 Hz
Y: -34.81 dBV
X: 3600 Hz
Y: -41.66 dBV
X: 4800 Hz
Y: -53.31 dBV
X: 9000 Hz
Y: -56.45 dBV
SPECTRUM ANALYSIS
ACCESS RADIO BROADCAST
Broadcast of 20 April
With BBC MICRO ON
and Colour TV ON
START: 0 Hz
BW: 95.485 Hz
STOP: 10 000 Hz


RANGE: -1 dBV
STATUS: PAUSED
A: MAG
-1
dBV
10
dB
/DIV
-81
X: 1250 Hz
Y: -4.47 dBV
X: 2500 Hz
Y: -23.06 dBV
X: 3775 Hz
Y: -27.88 dBV
X: 5025 Hz
Y: -35.17 dBV
X: 6275 Hz
Y: -43.83 dBV
X: 7525 Hz
Y: -44.99 dBV
Broadcast of 18 May
BBC MICRO OFF, TV OFF
START: 0 Hz
BW: 95.485 Hz
STOP: 10 000 Hz

SOME SAS CONCEPTS

GCF

If you spend time to understand the basic concepts involved, it will be much easier to follow the SAS "railroad" and other notations next month.

The fundamental concept is the frame - a straight steal from the solo adventure books, and also used in Microtext. There are some differences though, and several other key concepts, but they all need to be explained as part of the frame-work. Frames are the building blocks of this adventure system.

Some questions...
What is the difference between a room, and an object? The obvious reply is that a room cannot be picked up, and you cannot get inside an object - seems obvious! BUT in Level 9's "Dungeon Adventure" there is a packing case which you can TAKE around with you, which you can also go IN. It has several rooms inside it. What then, is the difference? None at all. The difference only lies in their use.

What is the difference between a {room with a fiendishly cunning trap} and a monster? Well you should be able to deduce the answer I'm expecting...

The point I'm making is that rooms, objects and monsters can all be referred to as frames. There can also be frames dealing with situations, such as choosing which spell to cast
(an example of a "situational" frame is VSAS 999 - lines 750-770 in the listing). There are other types of frames that can be used - for instance, to implement combat routines - but this should become more apparent later.

FRAME CONSTRUCTION

Frames vary in size and complexity; each part will be identified by its initial character, which is one of "#$%&". Full syntax and other examples next month.

1. Frame Header:
Every frame starts with one of these, even if they have no other parts. It contains:
   a) frame identification number,
   b) brief title.

What you can see in a room is generally a list of titles of all the visible items in the room, other than the ones you are holding

```
    eg  DATA #17,Control Panel
```

2. Conditional Description(s):

Frames can have zero or more of these. They are used when you enter a room or examine an object. Some description of the room should be given, even if you cannot see - eg "the room stinks" - you surely don't need a lamp on to smell? (Level 9's "Colossal Adventure" requires light to smell!; but try going down the well without the lamp on)
Additionally you may want to change the status of the room just entered (some events may only occur the first time you enter the room).

Thus each conditional description of a room has 3 components:
  i) condition which decides whether the rest of line is executed
  ii) the text of the description
  iii) any special activities required to be done. eg:

```
   DATA $T,the air smells foul,
   DATA $p9,you see a control panel,SC5
   DATA $c5,you hear a low-pitched hum,
```

N.B. "T" is a condition which is always TRUE
"p9" has been previously defined as TRUE only if you can see
"SC5" - "tell" the control panel there is a light on
(actually Set bit 5 of the Current room's status)
"c5" - once any light has fallen on it, the control panel will remain active, even if you come back later without your lamp.

3. Conditional Option(s):

Frames can have zero or more of these. They allow options to be given to the player only when they are valid - there is no point in asking someone to push the green button on the control panel, if they can't see it! All valid options are given to the player, and the game stops until one is selected. Then whatever activities are associated with the option are done.

The three components are similar to those for conditional descriptions:

```
    DATA %p9,push green button,SC7
    DATA %p9,push yellow button,SC7
    DATA %T,wait,CC7
```

N.B. if you can see the control panel you are offered the choice of two buttons to push. If any are pushed, the Status bit 7 is set, else it is unset.

4. Conditional Followups:

There can be zero or more of these. They are only checked after an option has been selected. They have similar components to conditional descriptions:

```
    DATA &c7,I'm sorry - but you shouldn't have pushed any buttons - the control panel just blew up and killed you!!!,AX
```

N.B. "c7" is true only if a button was pushed

"AX" indicates to the interpreter that the adventure has finished (Adventure eXit)

5. Status:

Technically this is a fifth part. All frames have 32 status flags (or bits) automatically assigned to them. Normally you don't have to do anything special with them unless you want to.

ACTIVITIES

The examples of frames have referred to activities (abbreviated to ACTs) - specifically to acts "CC", "SC", and "AX". There are about 20 other types of acts, and there can be several together if necessary. Wait til next month.

A very useful one is "ME" - Move to Enclose:

eg the option to put golden nugget (frame #51) into your backpack (frame #54) could be specified as:

```
    DATA %p9,Put nugget in pack,ME51/54
```

This can also be used to move a player (who is always frame #0) to another room (eg frame #37)

```
    DATA %T,climb ladder,ME0/37
```

CONDITIONS:

Ignoring the initial character ($,%, or &), they can be any expression that can be EVALuated, using any of the "variables" known to the interpreter. Even the RND function can be used (and SIN, COS if you can think up a daft enough excuse) eg:

```
    DATA $r9<RND(6),your lamp flickers,
```

N.B. "r9" - a variable defined to contain the number of turns you still have before your lamp runs out of oil or battery runs out of charge. (the assumption is that as your fuel runs low, the lamp is more likely to flicker)

VARIABLES:

There are a number of special variables known to the interpreter which can be used, like "nt" is Number of Turns elapsed since the start of the game.

See you next month...

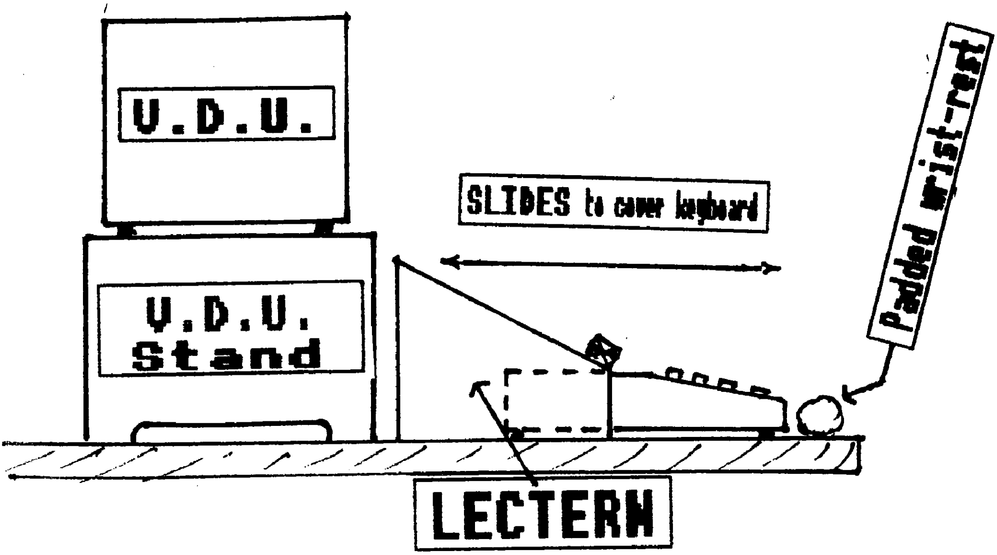
BBC LECTERN

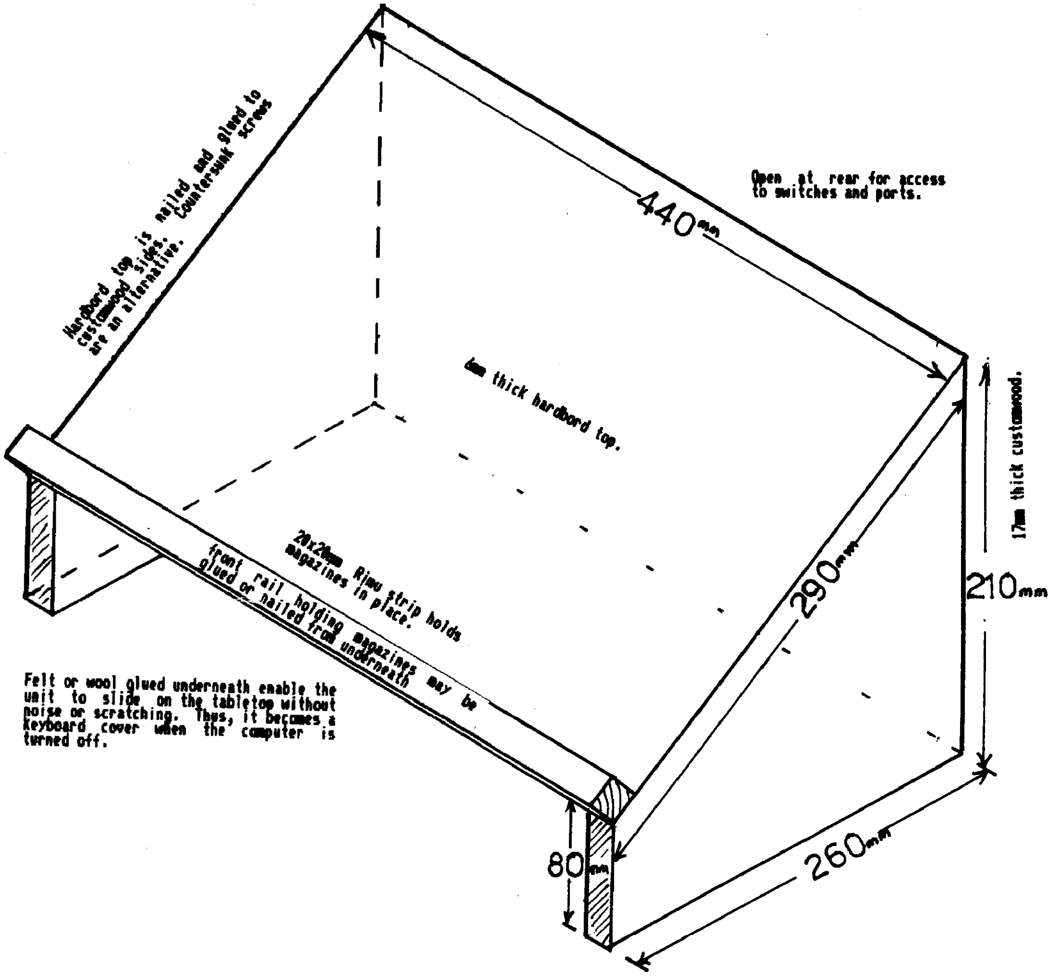
Alan Jackson - Oamaru

Cramped? Uncomfortable? Nowhere to put papers etc when using the Beeb? Alan has developed and tested (over some 18 months) a simple project to tackle these problems-

The monitor I use is raised up so that the centre of its screen is at the same height as my eyes when I am sitting up and making all those "correct posture " right angles with ankles, knees and lower spine. This also causes the monitor to be moved back some distance, just at arm's lenght, and left space for the bookrest (it also acts as a writing surface when the desk is covered with disc drives etc). The cables at the back of the Beeb can be led away under the bookrest and thus create space for all the othe bits and pieces on the bench. I am sure that the bookrest could be improved by the addition of some form of line guide to help when typing in listings from magazines, the one I used was so jerky in operation that I dicarded it after a short time.

Royden H., who kindly agreed to redraw the diagrams to improve their reproducability in Beeblet has also suggested that the front wooden strip used to hold magazines etc in place could be bach-cut to help hold the pages down. He also noted that he employs a padded wrist-rest : a stuffed sock.





Hardbord top is nailed and glued to customwood sides. Countersunk screws are an alternative.

Open at rear for access to switches and ports.

6mm thick hardbord top.

17mm thick customwood.

20x20mm Rimu strip holds magazines in place.

front rail holding magazines may be glued or nailed from underneath

Felt or wool glued underneath enable the unit to slide on the tabletop without noise or scratching. Thus, it becomes a keyboard cover when the computer is turned off.

ACID DROPS / BIRD STRIKE

Firebird.

Reviewed by David G. Auckland.

I actually didn't buy these two pieces of software from Britain, but rather it was a gift from a cousin who works in the office that one of the authors of these games works at. The games are put out by Firebird, which is British Telecoms own brandname for its venture into computing software. If the standard of these games is anything to go by, and if the prices remain as low as initially, then there will be large sales. (Great, but where can I get them, and for how much ? -Ed.). Perhaps this marketing attitude will encourage Acornsoft and others to drop the prices of their software by half. Perhaps this in turn will discourage piracy. Perhaps not.

However, on to the reviews. The first game is 'Acid Drops'. It comes in a normal cassette case, with a screen photo on the cardboard insert. Inside is the normal blurb to get you psyched up, and then come the more explicit instructions, with loading instructions directly beneath this.

The tape loaded with no trouble, apart from the obvious one with tape software - making a cup of coffee. The title page is attractive and informative - it includes an option to set joystick sensitivity: high, medium or low for those who are a bit extremist. The game itself is quite easy to master, but a lot harder to play and do well. The scenario is to shoot down the maurading moths with your laser base without being shot yourself. However, each shot that you fire puts a small hole in an acid bath suspended in midscreen. When the acid leaks out the problems really start. If you are hit by acid then you die - you only have five lives. If the moths get hit by acid they just turn meaner and metamorphise into pods and spinners (according to the instructions { you had me worried for a minute - Ed.}). However, once you dispose of the twenty or so moths, a gate to the right opens up, and your laser base takes off. The screen then scrolls to the right and you must steer the base through a maze to reach the next screen. This is an exact replica of the first, apart from the fact that the moths are meaner, more agressive and follow trickier flight paths.

Overall the game was a pleasant surprise and quite addictive to play - the only annoying thing being the ability of the moths to abruptly change flight paths and thus evade the laser beam you just sent to obliterate them.

The second game is 'Bird Strike'. The packaging is the same as for 'Acid Drops' and there are no problems loading it. The idea of this game is, first, to wing the planes which start at the top of the screen above the cloud layer, and come sweeping down. If you do manage to wing a plane, you then have to shoot down a pigeon which flies across the screen. If you succeed a note is added to the musical stave at the top of the screen. If you complete the tune then the music plays, you recieve a bonus of 1500 points and then move on to the next screen with planes of a more modern nature. If you hit the planes dead centre, rather than winging them, then they are distroyed. Each time their falling bombs hit you, you lose one life. You have three.

I personally preferred this game to the former, because of the little extra touches of care. When the program loaded a good rendition of 'A Long Way To Tipperary' played - shades of ANZAC day? Sound can be turned on and off during the game, and a pause can be taken on the way to achieving a high score. The screen is well drawn and the idea is original, although I question the morality of shooting poor defenceless carrier pigeons out of the sky (you did say this was produced by British Telecoms.... - Ed.) They say all is fair in love and war but isn't this too much!? Nevertheless, this game is more addictive than 'Acid Drops', is not so infuriating yet still manages to give a good game.

(Sorry, David, Ive already printed this month's Hall of Fame too many times - your score of 11,340 for 'Bird Strike' will have to go in next month..)

HINTS

David G. informs me that the hint on page 10 of the March Beeblet is inaccurate - sound can in fact be turned off using *FX210,1 as well *FX210,x where x is 3 or more.

Jane C. reports some difficulty getting comprehensive information regarding disc filing and accessing until discovering a book called "Filing Systems and Databases for the BBC Micro" by A.P. Stevenson - published by Granada, and reviewed in the February 1985 A&B Computing.

Tony W. writes that a useful addition to the January project (a book or magazine support from Jeff P.) is a "line magnifier" - a long narrow magnifying lens, usually with a flar graduated scale across the top, which can be bought cheaply from many stationers. Drill holes in each end and attach a small piece of thread - nylon is ideal. Attach a counter weight to the other end of the threads - eg a piece of dowell about the same weight as the lens - and pass the thread over top of the support board.

THE NEW MAGAZINE LISTINGS

Tim Ryan

Although LISTO 7 provides a simple way to do a tidy listing, it has one or two rather annoying habits. The most annoying, in my view, is that if there is an overflow from one line to the next, it starts in column zero, rather than at the same indent as the first part of the line. It also indents only two spaces, and I prefer three. The other problem caused by a bug in BASIC is that the indenter routine does not recognise the backslash as a comment and therefore recognises assembly listing comments as BASIC commands and indents appropriately.

To combat these problems and also simply as an experiment in programming I decided to write a lister of my own. The fruits of my labour are now (sometimes -Ed.) used in BEEBLET. As it can be a little confusing following the indenting if one is not used to it, I was requested to write a short article describing what it does.

The most confusing part is its indenting when a line overflows so here is an example of it in action, with an explanation of how to type it in. The original lines were:

```
10REPEAT
20PRINT "The rain in Spain falls mainly on the plain."
30UNTIL FALSE
40END
```

The number of columns is set to 37 normally in the mag, so LISTO 7 and WIDTH 37 would produce:

```
  10 REPEAT
  20   PRINT "The rain in Spain fall
s mainly on the plain."
  30   UNTIL FALSE
  40 END
```

This is quite hard to read as the indenting is destroyed by the overflow to the next line. My routine produces the following instead:

```
  10 REPEAT
  20   PRINT "The rain in Spain fall
        s mainly on the plain."
  30   UNTIL FALSE
  40 END
```

This, I feel, is much clearer, as the area of the loop is clearly defined by the indenting, with no parts starting at the left edge of the page to confuse things.

To type the listing shown immediately above, you should type the first listing shown. There is no need to insert any spaces at all, except the normal single space between words on the line.

To compare my listings against the standard LISTO 7, look at the May issue.
Pages 32 ff is one of my listings
Pages 11 ff is LISTO 7

Which do you find easier to read ? Which has clearer indenting ? The editor and I would both be most pleased to hear your comments, and (constructive) criticisms.

MALEBAG

Dere sir,
i red you're articel on page 12 and was most intarrested in the poins rased. it seems that yuo culd be waiting for a reply and as i am a silent mass i ams speeKing up on behalve of the vogel minority who might not otherwise say anything.You rased some intarresting numbers and it mite be in order if i answereed the m in the order so that it would be in order for reading the queerys and the answirs but not riteing them both.

1.done worry you'rel soon be able to save you're good money as well
2.Know exprets belong in this group ecxept one or too
3.very true!!!! sum of us done smoKe ether
4.you havnt read this magzine
5.ha ha i was rite wasnt i
6.wel thast one line b(ut and old one
7.any body who lives in the boondocKs needs to addres the problem . ever thought of ***
8 ,9, i have some adiquate replys for thees two but i se that you 're not goan too so i wont
0 im catcching on to this nombering thing but may get lost after this :doan seem to have left me any way into the h**adismals. wel thast rely all so now the vogal minority fals silent a gain

annymous.
p.S soryy about the speling the tWinK has all gone hard

A REFUTATION?

1. I am inclined to agree with the member who described publishing "Complaint" in full as an editorial aberration - I expected it to be edited, but it wasn't.
2. I have an "e" on the end of my name - Please keep it there. At times it has been the only way of distinguishing between several Ann/es at school, work, social gatherings...

NOW to Tim's points:

1. MONEY

Good money is NOT paid just for this magazine:

a) for your membership you also get access to decent cheap tapes and quality discs (who uses neither?), bulk purchase deals on drives, printers, monitors... the modem project...
b) with the help of local group committees, it tries to keep members informed of local events - cf Christchurch group's calendar in a recent issue; report on Copyright seminar et al.
c) the cover price of $3 is a non-member's price.
d) it is less than half what is paid for Elite - a decent game - and even more miniscule when compared with the cost of a single business program.
e) for many members, membership can probably be tax-deductible.

2. A LEARNER?

That is precisely who I think should be filling a large part of the magazine. For instance March'85 - Tape menu program. You can't get much more straighforward than PRINT statements, and here they couldn't be more effective!
Reread letters etc published in 'Beeblet'. It is the newcomers to computers that have a "sense of wonder", that are keen, excited, and in need of encouragement to maintain their interest while they attempt to overcome the hurdles of "Bad Program", Syntax error at...". Before graduating to the 1MHz bus. Surely there is a place for this in Beeblet, and for what may appear to be reinventing the wheel. (After all, who reads the User Guide?)

3. TOO SPECIFIC INTERESTS

A comment overheard was that Feb'85 teleprinter driver article was of such limited interest that it shouldn't have been published. A reaction to that was that whilst they didn't have a $10 printer, the program was interesting for the techniques used. In the questionaires returned, is a thank you from someone who is actually using the program. Similar positive reactions have been made concerning Relocating View -

superficially it is only of use to those who have squandered $1000-odd on a co-6502 on top of the cost of View.
Someone who didn't have disc drives made use of a concept presented in articles on them: MENUs, and applied it to tape (March '85).

4. TO EXPLAIN OR WRITE IS HARD

a) Quite a few of the articles under my name are in fact written explanations of verbal questions made of me at local meetings. (It's safe to open the pages of 'Beeblet', Ann S - your printer problem is in this category). I've tried to get these questionners to write up the answers themselves, because to explain is to relearn, or to learn more fully. Standard technique surely. Since this approach fails too often, I write and expand a little the 'answers' on the assumption that if one person has asked, and others listened, there are probably others who also want to know. So force yourself to write down things - it becomes easier with practice.

b) Having access to all the membership forms, I shall further comment:
i) Many members have tertiary, or professional qualifications of some sort - MB ChB, NZCE, Dip Tchg - or are studying for them. To acquire such levels of training implies some ability to express oneself on paper adequately.
ii) Many members also appear to be in, or approaching, management positions. Do they not have to present reports / explanations at work? What about the self-employed? Ever tried to explain to your accountant something tricky?
iii) Are there no parents in the User Group, who have tried to explain how to do things, to their children?
iv) If you have a physical handicap that prevents you from putting pen to paper, or using PRINT statements on the Beeb/Elk, why not talk aloud, with a microphone somewhere in the room. I'm sure the Editor would find someone to transcribe it for you.

5. TOO BUSY

"If you want something done, ask a busy person." Anyway, if you write for the magazine, you won't need the time to read it!?!

6. TOO YOUNG / OLD...

Youth: Hall of Fame, a lot of the reviews, an article on *LINE (when it appears)... What about "How I conned mum & dad into getting "x" for the computer" and see 7b below.

Age: John Bull Oct'84 p.7 - he's not only a grandparent, he's "isolated" on Waiheke Is. Jean's Interest and Wallpaper programs Dec'84. She's also retired.

Other handicap: David B is a registered blind person, with a Beeb who has
a) learnt to program
b) got Frank H, another member, to write some more "advanced" programs to assist him (good for Frank who also gives thanks to the Speech U.G.)
c) told us through the magazine about these things (Mch '85 esp).

7. NON-ARTICLE HELP:

a) A quote from Alan, in Oamaru,
"I have already used my Beeb to train teachers on a 1 to 1 basis on BBC computing in the subjects of their expertise. Now a few interested ones meet in each others homes to swop ideas and drink tea"
b) Sandra, age 16, travelled from Wainuiomata into and across Wellington to our place to spend a day transferring programs in last year's Beeblets from Beeb discs to tape and testing them on Whitcoull's Electron. That was some 4 hours driving for her dad in all.
c) For those of you who have been round long enough, remember "The Blessed Event" ? Issues 1&3 of Vol. 1. Also Napier Tony's garage sale - I just wish I had a garage to sell.
d) Chris and Jane, who run the Tauranga group, say everyone is new to programming and computers, and that they all find it hard going working from books. The advantage I see with a magazine is that you can start up a 'conversation' in its pages, or write in and ask for or offer elaboration on an article already published.
e) Volunteer to make up modems?? (see April mag)
f) Talk to your local dealers, find out who else around he has sold Beebs or Electrons to, and try to meet them.

g) If all else fails, become a civil servant and transfer to Wellington at the expense of other people's taxes - we could use you on the committee!!

NAILS ON THE HEAD?

My "Complaint" has apparently needled or prodded quite a few people into doing something "at last". There is some interesting material coming through, or promised. Quite a few people have admitted to needing the prod of my letter plus survey, to do something they "always intended doing anyway - some day".

I did not intend to sound "nasty" - I went through several drafts to avoid that - I was letting off steam built up when I was looking for a small item and could only find "Anne F"- or "GCF"-authored articles. I also thought the figures made interesting reading, and handed the whole lot to the Editor, to make what he would of it (if anything). I guess the fact that it was already in a Wordwise file on disc guaranteed it would get printed unaltered. That's life? (some things have more impact au naturele -Ed.)

GEOFF suggests that the Editor is short of material - he's not so short that he has to publish everything written by Gavin or me - let alone put them all in one issue. It's just that we give him a Wordwise file on disc, plus a printout, and that's easier for him than typing it in himself! (Actually, I have a bias towards anything supplied on tape or disc.. - Ed.)

FINALLY I shall look forward to seeing a whole host of new names, short programs to type in for fun, positive discussion of the direction of the User Group, and hope like mad that someone in Timaru, or Christchurch, or Hamilton, or wherever people get together to gossip about Beebs and Electrons, will write in, and make 'Beeblet' not just a NZ magazine, but a NZ-wide newsletter as well. After all, there are 40 pages. I personally think you need at least this many to ensure:

a) 'something for everyone' each month,
b) in-depth articles for those who complain of shallowness,
c) a variety of topics maintained,
d) room for local news items when they can be winkled out of people... what did happen at that session on robotics, mainlanders?

LEVEL 9

LEVEL 9 have released a new adventure "Emeral Isle", retailing at about 7 pounds (mail order), which is supposed to be a little simpler than their other offerings (it couldn't be harder!). It includes full colour graphics.

The June Acorn User notes that there is a new version of Elite out for the second processor. It also mentions that two more manufactures have launched products to compete with the AMX mouse : "Megamouse" from Wigmore House (88.5 pounds) and "Magicmouse" from SMC (whoever they are) for 55.95 pounds. Details of the BBC+, which is also "unveiled" by Neil Gordon in this issue, are provided too.

USING BBC BASIC by Roger McShane

227 pages, $26-00, Published by Prentice Hall.

Reviewed by Kay Eff Cee

When first given this book to review I eyed its bulk and thought "Oh yes a few day's reading." I soon discovered that it was not just a reading book. It is very much a doing book as are, no doubt many computing books. So my few day's reading became 2 months doing.

Each of the eighteen lessons contained in the book is built around four headings:

1. The lesson - which deals with a particular aspect of computing eg. keyboard, arrays.

2. Summary - in this section any new skills introduced in the lesson are highlighted.

3. Housekeeping - Comments on commands which may or may not have been used in teaching the principle skill in the lesson, but which you will ultimately need to know and understand. eg. in the Introduction to Graphics the author concentrates on the MOVE and DRAW commands. The CLG command is not used, but appears under housekeeping.

4. Exercises - these provide an opportunity to apply what has been learned in the lesson.

The book is written in simple language and is very easy to read. Each lesson is very much a small step by small step progression, so you get the picture whether or not you are being introduced to graphics!

If you prefer the school textbook layout and approach to learning complete with questions and answers you will probably find this book useful. If you can cope with the heavier bombardment of information as per the User Guide you will find the book a bit simplistic.

WHO GOT THEIR MONEY BACK?

The short answer is, we don't know. We offered prompt renewing members an opportunity to get their renewal money refunded. To totally avoid accusations of "cooking the books" we have decided to let the Golden Kiwi choose the winning member!

Below is a list of all the membership numbers of people who:
a) renewed before 30 April
b) filled in and returned the questionnaire by the same date
c) were not national committee members for the 1984/5 or 1985/6 years!

To find the lucky person, look for the winning ticket number of the Golden Kiwi lottery result published in major newspapers on Thursday 27 June (or the first day after that date that results are published). If there are two lotteries published on that day, then the winning ticket is of the first lottery number printed. (Substitute the winning ticket number for "KIWI" below)

Switch on your beeb, and type in:

```
PRINT KIWI MOD 132
```

The answer is the table-number referring to the membership number of the winner below. That person can expect a cheque in the mail shortly!

	0	1	2	3	4	5	6	7	8	9
0	112	113	114	116	118	120	121	122	123	125
10	140	142	145	146	148	153	154	157	164	165
20	178	180	183	198	200	202	212	215	226	227
30	233	236	239	240	245	246	248	249	250	254
40	260	263	269	281	282	292	295	299	306	307
50	322	329	337	339	344	345	352	359	365	372
60	380	381	385	386	388	389	391	392	396	403
70	404	407	408	409	414	428	429	437	439	445
80	446	449	452	455	457	460	461	464	465	467
90	468	469	471	472	475	479	484	493	497	498
100	503	504	506	507	514	516	518	519	521	522
110	526	528	532	536	537	542	544	547	548	549
120	551	553	554	557	560	564	565	571	584	592
130	595	599								

HALL OF FAME

PUBLISHER:	GAME	HISCORE	SCORER
Acorn	Arcadians	34,740	Rasik R.
	Aviator	5,610	Joanne C's Dad
	Countdown	246	Andrew P.
	Elite	Elite	J&C Climo*
	Meteors	46,200	Andrew P.
	Monsters	107,390	Robert T.
	Planetoid	413,875	Peter J.
	Rocket Raid	140,777	Mark R.
	Snapper	250,000	Susan
	Starship Command	9,495	Robert N.
	Super Invader	59,250	Kiran R.
?	Cylon Attack	157,970	Aaron
Alligat.	Bug Blaster	37,642	Steve W.
Aardvark	Zalaga	215,180	Peter J.
	Frak	1,026,820	Andrew P.
	Monsters	175,110	Geof P.
Amcom	Fortress	89,900	Andrew P.
Bugbyte	Twin Kingdom	385	A. Beeston
Comp.Con	Android Attack	236,195	Peter J.
Dr Soft	'747'	205	Steve W.
FG Soft.	Mutant Defender	6,900	Steve W.
	Lords of Time	925/1000	Steve W/Kerry K.
	Snowball	900/1000	Steve W./Dianne
Prog. Pow	Felix in Factory	12,250	Rasik R.
	Frenzy	915,468	Dianne T.
	Killer Gorilla	91,000	Susan
	Laser Command	109,225	P. Leadley
	Moon Raider	112,350	P.Leadley
Sup.Soft	Q*Bert	43,825	Peter J.
	Roadrunner	97,000	Andrew P.

* (disc) 2 tasks performed

HALL OF FAME

USER GROUP MEETINGS

AUCKLAND: -meets 2nd Wednesday of the month, 7.30pm, Conference Rooms, Auckland Secondary Teachers College,60 Epsom Ave, Epsom. ph Dave 770-630 x 510(bus) or Kerry 695-355.
-North Shore: meets last Wednesday of the month, 7.30pm, Salvation Army Hall, Glenfield Road, Glenfield ph Chris 419-0543 (wk)

CHRISTCHURCH: -fortnightly workshops, Monday 6.30pm at Hagley High School. ph. Michael 582-267.

DUNEDIN: -first Thursday of the month. Phone Martin McDowall 42-031

HAWKES BAY: -Alternative months Hastings & Napier. Contact Kendall Napier 435-624, Bob Taradale 446-955, Mitch Hastings 778-235.

INVERCARGILL: -Phone Evan 330-444 Invercargill.

PALMERSTON NORTH:-Phone Tom Skinner 83-009(wk)

TAUPO: -meets every second Wednesday @ 7pm. Contact David 84-215 (h), 85-215 (wk).

TAURANGA: -every second Wednesday. ph. Chris & Jane 65-076.

TIMARU: -Phone Lloyd van der Krogt Timaru 61-412.

TOKOROA: -meets first Friday of each month, Tokoroa High School, 8pm. Phone Brian 67-625 Tokoroa.

WAIKATO: -first Friday of the month 7.30pm Waikato Tech Institute. Ph. Alison or John, Morrinsville 6695.

WELLINGTON: -meets last WEDNESDAY of the month, 7.30pm, Royal Society Lecture Theatre, Turnbull St, Thorndon.Workshops: second Thursday of month, other venues. ph. Don 848-235 or Beryl 862-874 or Mike 785-437.

OTHER CENTRES: - let us know and we will publish details here.

THE NEWSLETTER:

CONTRIBUTIONS:-most welcome. Listings should be sent in on tape ordisc.(Tape - 2 copies, one at 300 baud please). Please include written explanation of listing; text and graphics only material welcome too.

DEADLINES:- For a particular issue, the last day for material is the last day of the month prior to publication.

ADVERTISING:- Rates are $20 per half page (camera ready), deadline as above.

Copyright (c) BBC/ACORN COMPUTER USER GROUP OF NZ INC.

Published monthly except January and mailed to financial members.

MEMBERSHIP:

Membership of the users group is on payment of an annual subscription from April to March. For the 1985/6 year it is: $30.00, plus $5.00 for Auckland & Wellington members as a local branch levy, which entitles attendance at meetings as detailed above. Renewing members get all back issues of the current year.

THE BBC/ACORN COMPUTER USER GROUP OF NZ INC, P.O.Box 9592, WELLINGTON.

Reproduced by ROBERTS PRINTING CO. LTD, WELLINGTON.